

Attorney Docket No.. D2538

PATENT
IN THE UNITED STATES PATENT & TRADEMARK OFFICE

Inventor: Alexander Medvinsky)	Confirmation No.: 8249
)	
U.S. Serial No.: 09/765,108)	Customer No.: 000043471
)	
Filed: January 16, 2001)	Art Unit: 2136
)	
)	Examiner: Carl G. Colin
)	
Title: SYSTEM FOR SECURELY COMMUNICATING INFORMATION PACKETS		

DECLARATION UNDER 37 C.F.R. § 1.131

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir,

I, Alexander Medvinsky, hereby declare as follows:

- I am the named and true inventor in the above referenced patent application and that I am the sole inventor of the subject matter disclosed and claimed in the above referenced patent application.

U.S. Serial No.: 09/765,108

2. I submitted a description of my invention, now claimed in claims 1-7 and 10-23 of the above application, to the law department of General Instrument Corporation in an "Invention Record Form." I signed the Invention Record Form on September 12, 2000 and the signatures on the Invention Record Form are my own. A redacted copy of the Invention Record Form is provided with this declaration as Attachment A which includes a copy of a paper describing the invention.

3. I conceived the invention recited in claims 1-7 and 10-23 of the above application prior to December 19, 2000. See, Attachment A, Invention Record Form

4. Upon information and belief, the stamp bearing the date of September 19, 2000 on the front of the Invention Record Form was placed thereon by the law department of General Instrument Corporation and indicates the date of receipt of the Invention Record Form.

5. Upon information and belief, on or about October 10, 2000, a description of my invention was forwarded to the law firm of Townsend Townsend & Crew at which a patent application was prepared. See, Attachment B, Letter Dated October 10, 2000

6. I hereby declare that all statements made herein based upon knowledge are true, and that all statements made based on upon information and belief are believed to be true, and further, that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Dated: 9-8-06

By: Alexander Medvinsky
Alexander Medvinsky

ALEXANDER MEDVINSKY DECLARATION ATTACHMENT A

INVENTION RECORD FORM

Invention Record Form
GI Docket No. D2538

I. Administrative Information

1. Short Descriptive Title of the Invention:

RC4 Encryption for RTP with CODEC Changes and Collision Resolution

2. Identify all persons who contributed to this invention, including persons from other divisions and/or outside companies:

	Inventor 1	Inventor 2
Full Legal Name	Alexander Medvinsky	
Home Address	8873 Hampe Court	
City, State, Zip	San Diego, CA 92129	
Citizenship	United States	
Division/Co. Location	Advanced Technology Dept. Motorola BCS San Diego	
Office Phone No.	858-404-2367	
Mgr.'s Name & Phone No.	Eric Sprunk 858-404-2426	
Signature of Inventor	<i>Alexander Medvinsky</i>	
Date	9-12-2000	

	Inventor 3	Inventor 4
Full Legal Name		
Home Address		
City, State, Zip		
Citizenship		
Division/Co. Location		
Office Phone No.		
Mgr.'s Name & Phone No.		
Signature of Inventor		
Date		

3. ☐ Check box if there are additional inventors listed on separate sheets. Additional information concerning inventors, if any.

Invention Record Form

II. Background Information

1. Do you believe this invention was developed while working under or in the performance of experimental, developmental or research work called for by a government contract or with the understanding that a government contract would be awarded? ☒ No ☐ Yes If yes, please explain:

2. Has your invention been disclosed to anyone outside General Instrument in a speech, exhibit, presentation, product, product manual, report, lecture, trade show, technical article, publication or otherwise? ☐ No ☒ Yes If yes, please explain:

It was presented to PacketCable standards group for IP Telephony over Cable.

3. Is this invention related to any previous GI invention disclosures of which you are aware (made by you or someone else)? ☒ No ☐ Yes If yes, please explain:

4. Name of product(s) and/or project(s) for which this invention was developed:

This is part of our contribution to the PacketCable security standard.

5. Planned or actual use of invention:

Will implement inside our MTAs (Multimedia Terminal Adapters), which are IP Telephony clients.

6. What economic benefits do you think GI can derive from this invention?

RC4 encryption is low-cost, fast and easy to implement. This invention makes RC4 encryption applicable to RTP by fixing some of the problems associated with media streams over IP.

7. When do you expect a product incorporating this invention to be sold, offered for sale or shown to someone outside of GI? (If a product or prototype has already been sold, offered for sale or shown, please identify the earliest date this happened.)

In 2nd quarter, 2001.

8. Has a working model of the invention been built and tested (or appropriate software been written)? ☒ No ☐ Yes If yes, who has witnessed a demonstration, and when?

9. List below any patents, publications, articles, texts, products, etc. which describe technology similar to your invention including reference material which may be useful in understanding the background technology of your invention. (Use a separate sheet if necessary and attach a copy of each item. Please include copies of all bibliographical information) (Use a separate sheet if necessary)

The attached email from me on November 22, 1999 outlined the technique of re-deriving RTP encryption keys when the RTP timestamp wraps around. This particular disclosure is re-using the same technique to solve additional RC4 encryption problems that were not considered at that time.

Note that the technique described in the email was originally a result of a joint discussion at PacketCable. This disclosure doesn't cover that particular email. Instead, it covers the application of that same technique to solve additional RC4 encryption problems for RTP streams.

Alexander Medvedsky
Signature of Submitter(s)

[Signature]
Read and understood by [Witness Signature(s)]

9/12/00
Date

9/12/00
Date

b2538

Invention Record Form

III. Description of the Invention

1. Please provide a very brief (i.e., one short sentence) summary of your invention.

Provide secure RC4 encryption for RTP that allows for CODEC changes and resolution of SSRC (RTP source identifier) collisions.

2. Briefly describe the field of technology to which your invention relates.

This invention applies to RC4 encryption of any RTP streams.

3. Briefly describe the problems, issues or needs which led to the invention

RC4 is a stream cipher, where a random stream of bytes (key stream) is continuously generated and XOR-ed with the cleartext data to generate ciphertext (encrypted stream). There is a rule that the same portion of the key stream must not be reused to encrypt (XOR with) multiple messages. If this rule is not followed, the RC4 encryption could be more easily broken.

Within PacketCable, RTP timestamps are used as a pointer (synchronization source) into the RC4 random stream of bytes. Therefore whenever a timestamp is re-initialized, a new key stream has to be initialized, in order to avoid the re-use of the same key stream bytes.

Specifically, for audio streams we have:

$$\text{RC4 Key Stream Offset} = \text{FrameNumber} * \text{FrameSize}$$

The frame number is number of the audio frame generated since the start of the stream and can be derived directly from the RTP timestamp:

$$\text{FrameNumber} = (\text{RTP Timestamp} - \text{RTP Initial Timestamp}) / \text{Nu}$$

where Nu is the number of audio samples in an uncompressed frame of audio

After a CODEC change, FrameSize changes and the same formula can no longer be used to determine the RC4 key stream offset. PacketCable specification attempted to solve this problem by re-adjusting the FrameNumber right after the code change as follows:

$$\text{FrameNumber_new} = \text{rool} ((\text{FrameNumber_old} + 1) * (\text{FrameSize_old} / \text{FrameSize_new}))$$

After this adjustment, the frame number will continue incrementing by 1

It turned out that $(\text{FrameNumber_new} - \text{FrameNumber_old})$ depends on the value of FrameNumber_old and if the codec change is not recognized at the same time on both endpoints, they will adjust their FrameNumber by a different offset. Since RTP is sent over UDP and some of the packets might be lost in transit, there is no guarantee that FrameNumber_old during this adjustment will be the same on both sides.

Therefore the PacketCable solution does not appear to work for Codec changes, which led to this invention.

Alexander Medvedsky
Signature of Submitter(s)

John J. O'Brien

Read and understood by [Witness Signature(s)]

9/12/00
Date

9/12/00
Date

Invention Record Form

D2538

4. How have others addressed these problems, issues or needs?

Not that I know of.

5. Describe those particular features or functions of your invention which you think may be novel or technical advancements over the technology you listed in section II.9.

In II.9, I referenced an email that allows for generation of new, non-repeating RC4 key streams when an RTP timestamp wraps around. A similar technique is re-used in this invention but for the new purpose of handling codec changes and RTP Source ID collisions without compromising the security of RC4 encryption.

6. Best Mode: Describe any and all preferences you personally have regarding how to best implement, build, produce or use your invention (e.g., preferred parts, materials, techniques, etc. which you feel are best in practicing your invention). Each submitter's opinion is important here, even if there is disagreement. Please list anything you think will make the invention better in any way.

This invention does not contain multiple modes.

7. Briefly describe any alternative uses, variations or modifications of your invention which you contemplate.

This invention was initially specified for RTP packets containing audio frames that use RTP timestamps as the RC4 synchronization source. In general, this invention applies to:

- RTP encryption with any stream cipher that does not allow repetition of the key stream
- RTP packets containing any content, including audio, video and event packets
- An appropriate stream cipher synchronization source that is not necessarily the RTP timestamp. For example, the synchronization source could be the RTP sequence number or some other value inside the RTP header extension

If CODEC changes or RTP Source ID collision resolution require the synchronization source to be re-started, this technique can be applied to re-derive a new set of keys, to avoid a re-use of the same key stream.

8. Please provide any additional information you think should be known by the attorney reviewing this form.

In an attached email from David Lide of Telogy, he said that they agree with the solution for mid-stream codec changes and would have proposed the same thing.

9. Please provide a detailed description of your invention. Your description should ideally provide as many details of your invention as possible in order to achieve optimal patent protection. An ideal disclosure should describe the construction and operation of the invention including drawings (flow charts, schematics, block diagrams, mechanical drawings, photographs, etc.) and any relevant engineering laboratory notebook pages, reports, program listings, etc. If you have already prepared reports or other descriptive information, there is no need to rewrite it. Simply attach it and reference it in your invention disclosure data sheet (for example, "see attached 9 page engineering progress report addressed to John Doe dated 1 Jan., 1992 for description of amplifier circuit").

The invention is described in the attached document, labeled "Problems with I01 RC4-based Encryption and Proposed Solutions", dated on 9/5/2000. In particular, the invention is described in sections 3.1 and 4.1.

The solution in this document refers to a pseudo-random function $F(S, \text{label})$. This function F generates a new pseudo-random set of bytes for each unique pair of values (S, label) . The output of F is used as an RC4 encryption key. Any such pseudo-random function can be used – for example by applying a SHA-1 hash to the concatenation of S and label.

Alexander Vlodavsky

Signature of Submitter(s)

John F. Zelen

Read and understood by [Witness Signature(s)]

9/12/00

Date

9/12/00

Date

9/5, '2000 - Dave. J. Roy

D2538

Problems with I01 RC4-based RTP Encryption and Proposed Solutions

By Sasha Medvinsky, Motorola

1 Introduction

This memo assumes that RC4 is the required encryption algorithm and will remain the required encryption algorithm in the near future. Therefore, it proposes solutions specific to RC4 and does not explore the possibilities of using a block cipher instead. Alternatives to RC4 that would provide sufficient performance for RTP encryption will be explored separately and are outside of the scope of this memo.

2 Support for RFC 2833 (AVT Tones)

RFC 2833 requires support for events that span multiple RTP packets. In those cases, multiple consecutive RTP packets are required by this RFC to contain the same RTP timestamp – the one corresponding to the start of the event.

The I01 PacketCable security specification assumes that each RTP packet contains a new timestamp, where a timestamp is always incremented by a multiple of the audio frame size (measured in terms of audio samples). The specification does not disallow RTP event packets, but it does require that the timestamp is incremented for each event packet the same way as it is done for normal audio codec packets. This clearly conflicts with RFC 2833, which requires support for multiple event packets with the same timestamp.

2.1 Historical Overview of Previously Discussed Solutions

The reason that the timestamp has to be unique for each RTP packet for the I01 security specification, is that it is used as a pointer into the RC4 key stream (which is XORed with the media stream to produce encrypted packets). It is generally considered insecure for a stream cipher to re-use the same key stream bytes to encrypt multiple packets.

On the other hand, it is highly desirable to use the timestamp as a pointer into the RC4 key stream. One alternative would be the RTP sequence number, which has the following problems:

- a) It is too small, only 2 bytes, and might wrap around in the middle of a call – as often as several times an hour depending on the codec. The MTA would have to keep track of the number of times the sequence number wraps around and could possibly miss one of those wrap arounds and lose sync.
- b) The RTP header is only optionally authenticated (with MMH MAC) due to bandwidth considerations. If someone were to maliciously change the sequence number, it would constitute a denial of service attack. For example, the sequence number changed to 0xffff would result in the receiver generating a large number of key stream bytes, trying to catch up with that sequence number.

On the other hand, the security specification recommends that a sanity check is performed on the RTP timestamp – to make sure it is within an expected window, calculated based on the MTA's running clock.

One solution previously considered by the security team but apparently flawed was to use the combination of timestamp and sequence number into the key stream. In other words, when multiple packets are sent with the same timestamp, the timestamp points to a block of key stream bytes used to encrypt the whole

group of packets and the sequence number points to the correct place within that block for encrypting each individual packet

The problem with this approach is that the sequence number does not tell you where you are within a particular timestamp (when the same timestamp is used on multiple packets). For example, let us say that two consecutive RTP event packets are sent with the same timestamp, the 1st one was lost and the 2nd one was received. How does the receiver know that the lost RTP packet had the same timestamp? It can't possibly know, but it needs to know in order to find the correct place in the RC4 key stream

2.2 Proposed Solution for RFC 2833 Support

I propose to use the RTP header extension (as specified by RFC 1889). This extension would be used only for consecutive RTP packets that contain the same timestamp. In those cases, **the extension would contain the time interval from the beginning of the 1st packet with that same timestamp. The time interval would be expressed in terms of the number of audio frames (based on the current codec).**

When a timestamp is used for the very first time, this extension will not be used. When a 2nd RTP packet is generated with the same timestamp, it will contain this extension. Since for PacketCable, audio packets contain 1, 2 or 3 frames, the value of this extension for the 2nd packet will be 1, 2 or 3. The exact format of this extension is TBD.

The 1st RTP packet with a new timestamp does not contain this RTP extension in order to conserve bandwidth. It is assumed that all audio packets will have a new timestamp, while event packets will normally be smaller than audio packets (and thus the overhead associated with the extension will be absorbed by the smaller event packet size). Some codecs however could produce very small audio packets and in those cases the use of event packets would still require allocation of additional bandwidth.

More specifically, the I01 security specification defines the variable N_k , a byte offset into the RC4 key stream as follows:

$$N_k = N_f * (N_e + N_m)$$

Here, N_f is the count of the audio frames generated so far, N_e is the max number of event bytes that can be generated within a duration of one audio frame and N_m is the MAC size (which could be 0).

And, N_f is calculated as follows:

$$N_f = (\text{timestamp} - \text{RTP Initial Timestamp}) / N_u$$

Here, N_u is the number of speech samples in one frame of uncompressed audio.

The proposal in this section does not change any of the above calculations. Instead, it proposes that the timestamp used to calculate N_f is defined as follows:

$$\text{timestamp} = \text{RTP Timestamp} + \text{extension} * N_u$$

Here, RTP timestamp is the actual timestamp in the RTP header and 'extension' is the value of this new extension in the RTP header. If this extension is not present in the header, a value of 0 is assumed.

This proposal assumes that at the same time as event packets are generated, no audio packets will be sent out. This rule is not apparent from reading RFC 2833, however if audio and event packets are allowed to be sent out at the same time it creates bandwidth allocation problems. It appeared from the last PacketCable architecture call that no one was in favor of that approach.

3 Mid-Stream Codec Changes

The I01 specification defines the procedure for adjusting the RC4 key stream during codec changes, which is apparently flawed. The main issue during codec changes is that the audio frame size can change. The

RTP timestamp is used to determine the number of audio frames processed so far and is thus required to be a multiple of the audio frame size (plus a random initial value)

After a codec change, all of a sudden the RTP timestamp is no longer a multiple of a new frame size. The I01 specification provides a formula for adjusting the timestamp, so that it will continue to be a multiple of the new audio frame size

It turns out that the adjustment value added to the timestamp depends on exactly which audio frame is being processed when the codec change is discovered. However, there is no guarantee with NCS signaling that the two communicating endpoints will be notified (by their CMS) of the codec change at exactly the same time. Thus, there is a pretty good chance that after the codec change the two MTAs will lose synchronization on their RC4 key streams and all RTP packets will decrypt to garbage

3.1 Proposed Solution for Mid-Stream Codec Changes

The I01 security specification (updated by ECNs) currently provides for re-derivation of RTP keys when a time stamp wraps around. In that case, the key derivation function is:

$F(S, \text{"End-End RTP Key Change <N>"})$

where N is a counter that holds the number of times that the time stamp had wrapped around. (For the specification of function F() and parameter S, please see the I01 security specification)

I propose that we redefine the meaning of counter N. We can say that N is incremented whenever a new set of RTP keys has to be re-derived for the same media stream session. And a new set of RTP keys has to be re-derived when:

- a) RTP timestamp wraps around
- a) timestamp is re-initialized (after a codec change)

When the codec changes, rather than adjusting the RC4 key stream, simply generate a new set of keys by re-executing the above key derivation function and start a whole new RC4 key stream. Because now N is incremented with each codec change, a new pseudo-random set of keys will be re-derived after each codec change

4 SSRC Collision Resolution

RFC 1889 requires that each endpoint generating RTP session identifiers (SSRC) allows for the scenario when two identical SSRCs collide at a mixer or a bridge. RFC 1889 also requires that during such a collision, an RTCP BYE message is used to hang up one of the RTP sessions and that a new one is restarted with a new SSRC value.

It is unclear as to what happens to the RTP sequence number and timestamp when the RTP session is restarted after a BYE message. It appears possible that at least in some implementations the sequence numbers and the timestamp sequence are both re-initialized. Again, for security reasons when RC4 is used it is not acceptable to re-use the same RC4 key stream and re-start with the same initial timestamp value.

4.1 Proposed Solution for SSRC Collision Resolution

The proposed solution to this problem is very similar to the one proposed for the codec changes. Again, the counter N in the derivation function is incremented whenever a new set of RTP keys has to be re-derived for the same media stream session. And a new set of RTP keys has to be re-derived when:

- b) RTP timestamp wraps around
- c) codec changes
- d) timestamp is re-initialized (after a codec change or after an SSRC collision resolution)

Note that if the timestamp is not re-initialized after an SSRC collision there is no problem that needs to be fixed – the same RC4 key stream can continue to be used

5 MAC Size Changes

The security team last approved that to support MAC size changes for RTP traffic, the key derivation function would include the RTP ciphersuite (including the MAC used) and during the MAC change new keys will be re-derived.

While this method appears to work, since we are already using a counter to re-derive keys for other reasons (see above) To simplify the security specification we could also increment that same counter during the MAC size change

ALEXANDER MEDVINSKY DECLARATION ATTACHMENT B

LETTER TO OUSIDE COUNSEL



MOTOROLA

October 10, 2000

Charles J. Kulas, Esq.
Townsend Townsend & Crew
Two Embarcadero Center
8th Floor
San Francisco, CA 94111-3834

via FEDERAL EXPRESS

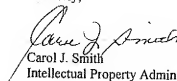
RE: NEW PATENT APPLICATION for
"RC4 ENCRYPTION FOR RTP WITH CODEC CHANGES AND COLLISION
RESOLUTION"
Our File: D2538

Dear Charlie:

Enclosed please find an Invention Record Form for a new invention. Please prepare a patent application based on the above identified invention disclosure. It is desired to file this application in the United States and Canada only as soon as possible. Please review the disclosure materials, attempt to draft some preliminary broad claims and contact the inventor(s).

If you have any questions, please contact me at (215) 323-1237.

Sincerely,


Carol J. Smith
Intellectual Property Administrator

Enclosure

D2538

FedEx Ship Shipment Receipt

From: Carol Smith
(215) 323-1237
General Instrument
101 Tournament Drive
Horsham, PA 19044

To: Charles J. Kulas, Esq
(415) 576-0200
Townsend and Townsend and Crew
Two Embarcadero Center
8th Floor
San Francisco, CA 94111-3834

COD Return Address:
N/A

Date: 10OCT00
Track Number: 791869416546
Service: Priority Overnight
Packaging: FedEx Letter
Special Handling: Drop
Piece: 1 of 1
Weight: 1 LBS
Dimensions: N/A
Declared Value: N/A
Deliver without Signature: No

Billing: Bill Sender
Bill To Acct: 217633729
Rate Quote: \$5 31
Reference: 50320

COD Shipment: No
COD Amount: N/A
Secured Check: N/A
Include Freight: N/A

Document Shipment: N/A
Commodities: N/A
Total Customs Value: N/A
Currency: N/A
Countries of MFG: N/A
Export License: N/A
Expire: N/A
License Exception Symbol: N/A
ECEN: N/A
Ultimate Destination: N/A

TERMS AND CONDITIONS

For complete terms and conditions see the FedEx Ship License

Agreement to Terms. By giving FedEx Your shipment, You agree to be bound by the terms and conditions specified in this document, the FedEx Service Guide and the FedEx Ship License. You previously executed, all of which are incorporated herein by reference for carriage of the shipment via FedEx delivery services to destinations located outside the United States. If there is a conflict between this document and the FedEx Ship License, the FedEx Ship License ("Service Guide") or the Standard Conditions of Carriage (which are available upon request from FedEx) then in effect the Service Guide or Standard Conditions will control, as applicable.

Customs Clearance. You hereby appoint FedEx as Your agent solely for the performance of customs clearance and certify FedEx as the principal consignee for the purpose of designating a customs broker to perform customs clearance. In some instances local activities may require additional documentation confirming FedEx's appointment. It is Your responsibility to provide proper documentation and confirmation, where required.

You are responsible for and warrant compliance with all applicable laws, rules and regulations, including but not limited to, customs laws, import and export laws and government regulations of any country to, from, through or over which your shipment may be carried. You agree to furnish such information and complete and attach to this shipment each document or submit shipment data to FedEx, as necessary to comply with such laws, rules and regulations. FedEx assumes no liability to You or any other person for any loss or expense due to Your failure to comply with this provision.

Letter of Instruction. If You do not complete all the documents required for carriage or if the documents submitted are not appropriate for the service or destination requested, You hereby instruct FedEx, where permitted by law to complete, correct or replace the documents for You at Your expense. However, FedEx is not obligated to do so. If a substitute form of air waybill is needed to complete delivery of Your shipment and FedEx's compliance that document, the terms of the FedEx Ship License and this document will continue to govern. FedEx is not liable to You or any other person for FedEx's actions on Your behalf under this provision.

Export Control. You authorize FedEx to act as forwarding agent for You for export and customs purposes. You hereby certify that all statements and information contained on all air waybills and SEDs relating to exportation are true and correct. You further certify that all Commercial Invoice information submitted with FedEx Ship is true and correct. You expressly authorize FedEx to forward all information of any nature regarding any shipment to any and all governmental or regulatory agencies which request or require such information. You acknowledge that civil and criminal penalties, including forfeiture and rate may be imposed for making false or fraudulent statements or for the violation of any United States laws on exportation, including but not limited to, 18 U.S.C. § 368, 22 U.S.C. § 401, 18 U.S.C. § 1901, and 50 U.S.C. App. 2401. You acknowledge that this shipment is not being sent to any entity listed on the Department of Commerce's Denied Parties List 15 C.F.R. Part 764, Supp. 2 or the list of Special Designated Nationals as published by the Office of Foreign Assets Control of the U.S. Department of the Treasury.

Items Not Acceptable for Transportation. FedEx will not accept certain items for carriage and other items may be accepted for carriage only to limited destinations or under restricted conditions. FedEx reserves the right to reject packages based upon these limitations or for reasons of safety or security. You may consult the FedEx Service Guide or Standard Conditions of Carriage for specific details.